



## COURS DE STRUCTURE MACHINE

### SÉRIE 06

#### OBJECTIF PÉDAGOGIQUE :

À la fin de cette dernière série, le stagiaire doit être capable de :

- Connaître les techniques d'échange de données.
- Connaître les différentes liaisons de transmission de données

#### PLAN DE LA LEÇON :

#### INTRODUCTION

#### I- TECHNIQUES D'ÉCHANGE DE DONNÉES

#### II- TECHNIQUES D'ÉCHANGE DE DONNÉES

- 1- Échange programmé
- 2- Échange direct avec la mémoire

#### III- COMMUNICATION ET SYNCHRONISATION

- 1- Liaison parallèle
- 2- Liaison série

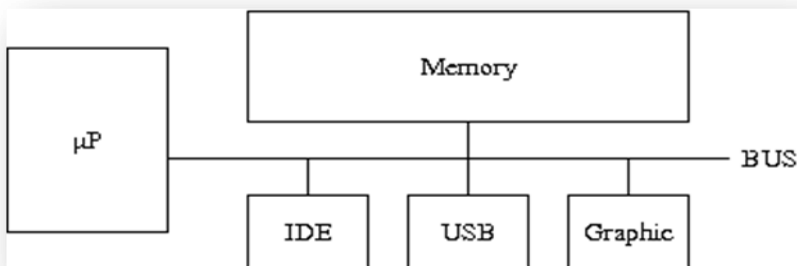
#### EXERCICE D'APPLICATION

## INTRODUCTION :

Un micro-processeur ne communique pas directement avec l'extérieur. Toutes les entrées/sorties se font par l'intermédiaire de circuits spécialisés qui déchargent ainsi le processeur de certaines tâches. Il existe des circuits spécialisés dans les différents types de connexions : série, parallèle, SCSI, IDE, USB, .... Un circuit pour une connexion série se charge par exemple de surveiller les signaux de la ligne et de le traduire en un octet disponible pour le processeur.

## I- TECHNIQUES D'ÉCHANGE DE DONNÉES :

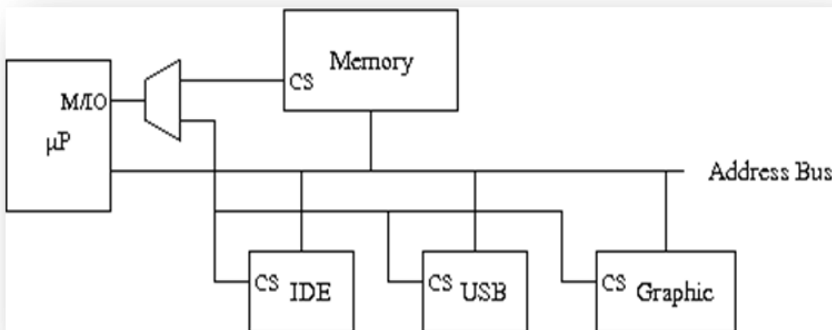
Ces différents circuits résident sur la carte mère et communiquent avec le processeur par l'intermédiaire du bus de donnée. La communication est en général à l'initiative du micro-processeur. Hormis les exceptions, ces circuits se contentent de répondre aux demandes du micro-processeur et d'exécuter ses instructions. Les circuits les plus simples ont généralement un *registre de contrôle* et un *registre de donnée* qui peuvent être lus ou chargés par le micro-processeur. Le registre de contrôle permet au micro-processeur de connaître l'état du circuit et éventuellement de le configurer. Le registre de donnée permet l'échange des données en entrée et/ou en sortie. Le micro-processeur lit dans ce registre les données en entrée et y écrit les données en sortie.



**Figure 1 : Circuits d'entrées/sorties**

Afin de répondre aux demandes du micro-processeur, les circuits doivent être sélectionnés par celui-ci. Il existe deux méthodes classiques pour faire cet adressage. La première méthode était plutôt utilisée par les processeurs anciens alors que la seconde est plus fréquente sur les micro-processeurs modernes.

La première méthode consiste à séparer l'adressage de la mémoire et l'adressage des entrées/sorties. En plus du bus de d'adresse, il y a une ligne supplémentaire où le processeur indique s'il s'adresse à la mémoire ou aux circuits d'entrées/sorties. Cette méthode a l'avantage d'augmenter d'une certaine façon l'espace d'adressage. C'est pourquoi les premiers micro-processeurs l'utilisaient car leur espace d'adressage était souvent réduit. Cette méthode a par contre l'inconvénient de nécessiter des instructions spécifiques pour accéder aux circuits d'entrées/sorties.



**Figure2 : Adressage distinct des circuits d'entrées/sorties**

La seconde méthode consiste à réserver une partie de l'adressage de la mémoire aux entrées/sorties. On parle d'*entrées/sorties en mémoire*. Cette méthode réduit donc l'espace d'adressage disponible pour la mémoire mais ce n'est pas gênant sur les micro-processeurs modernes qui disposent d'un (très) large espace d'adressage. Elle permet aussi l'utilisation des instructions de chargement et rangement pour accéder aux circuits d'entrées/sorties et de bénéficier ainsi des différents modes d'adressage. Cette méthode est toujours utilisée sur les micro-processeurs RISC car elle diminue le nombre d'instructions et simplifie le décodage de celles-ci.

## II-TECHNIQUES D'ÉCHANGE DE DONNEES :

Avant d'envoyer ou de recevoir des informations, le microprocesseur doit connaître l'état du périphérique. En effet, le microprocesseur doit savoir si un périphérique est prêt à recevoir ou à transmettre une information pour que la transmission se fasse correctement. Il existe 2 modes d'échange d'information :

- **Le mode programmé par scrutation** ou interruption où le microprocesseur sert d'intermédiaire entre la mémoire et le périphérique
- **Le mode en accès direct à la mémoire** ( DMA ) où le microprocesseur ne se charge pas de l'échange de données.

### 1- Échange programmé :

#### a) **Scrutation** :

Dans la version la plus rudimentaire, le microprocesseur interroge l'interface pour savoir si des transferts sont prêts. Tant que des transferts ne sont pas prêts, le microprocesseur attend.

L'inconvénient majeur est que le microprocesseur se retrouve souvent en phase d'attente. Il est complètement occupé par l'interface d'entrée/sortie. De plus, l'initiative de l'échange de données est dépendante du programme exécuté par le microprocesseur. Il peut donc arriver que des requêtes d'échange ne soient pas traitées immédiatement car le microprocesseur ne se trouve pas encore dans la boucle de scrutation. Ce type d'échange est très lent.

#### b) **Interruption** :

Une interruption est un signal, généralement asynchrone au programme en cours, pouvant être émis par tout dispositif externe au microprocesseur. Le microprocesseur possède une ou plusieurs entrées réservées à cet effet. Sous réserve de certaines conditions, elle peut interrompre le travail courant du microprocesseur pour forcer l'exécution d'un programme traitant la cause de l'interruption.

Dans un échange de données par interruption, le microprocesseur exécute donc son programme principal jusqu'à ce qu'il reçoive un signal sur sa ligne de requête d'interruption. Il se charge alors d'effectuer le transfert de données entre l'interface et la mémoire.

## **2- Échange direct avec la mémoire :**

Ce mode permet le transfert de blocs de données entre la mémoire et un périphérique sans passer par le microprocesseur. Pour cela, un circuit appelé contrôleur de DMA (Direct Memory Access) prend en charge les différentes opérations.

Le DMA se charge entièrement du transfert d'un bloc de données. Le microprocesseur doit tout de même :

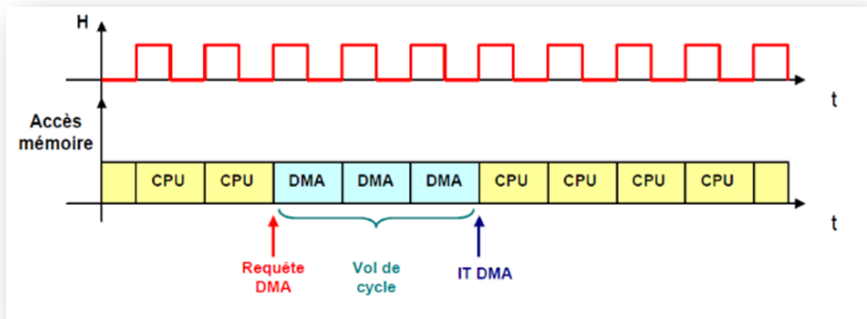
- Initialiser l'échange en donnant au DMA l'identification du périphérique concerné
- Donner le sens du transfert
- Fournir l'adresse du premier et du dernier mot concernés par le transfert.

Un contrôleur de DMA est doté d'un registre d'adresse, d'un registre de donnée, d'un compteur et d'un dispositif if de commande (logique câblée). Pour chaque mot échangé, le DMA demande au microprocesseur le contrôle du bus, effectue la lecture ou l'écriture mémoire à l'adresse contenue dans son registre et libère le bus. Il incrémente ensuite cette adresse et décrémente son compteur.

Lorsque le compteur atteint zéro, le dispositif informe le processeur de la fin du transfert par une ligne d'interruption.

Le principal avantage est que pendant toute la durée du transfert, le processeur est libre d'effectuer un traitement quelconque. La seule contrainte est une limitation de ses propres accès mémoire pendant toute la durée de l'opération, puisqu'il doit parfois retarder certains de ses accès pour permettre au dispositif d'accès direct à la mémoire d'effectuer les siens :

Il ya apparition de vols de cycle.

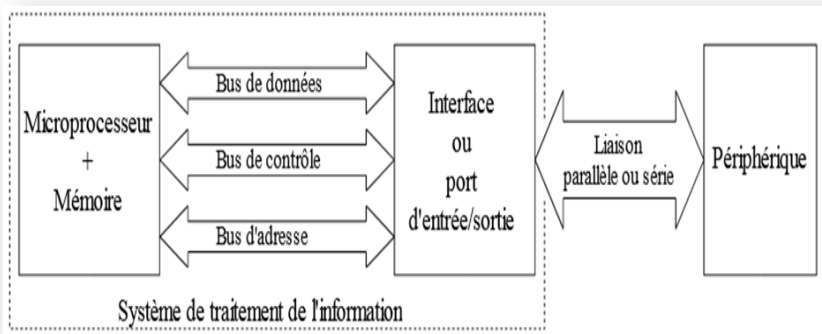


### III-COMMUNICATION ET SYNCHRONISATION :

Les systèmes à microprocesseur utilisent deux types de liaison différente pour se connecter à des périphériques :

- Liaison parallèle
- Liaison série

L'interface ou le port d'entrée/sortie est la fonction qui permet de transférer les données entre le système de traitement et un périphérique (imprimante, écran vidéo, préactionneurs, clavier, capteurs, boutons, ... ).



Le microprocesseur écrit ou lit le contenu du registre de données dans l'interface qui se charge de la communication avec le périphérique. Les bits des mots de données (8 bits) inscrits dans le registre de données peuvent alors être transférés par une liaison parallèle ou série.

L'interface est connectée aux trois bus :

- Le bus de données véhicule la donnée fournie par le microprocesseur vers le PIO ou du PIO vers le microprocesseur,
- Le bus d'adresse sélectionne un port parmi d'autre,
- Le bus de contrôle valide le fonctionnement du circuit et indique le sens de transfert (lecture ou écriture).

### **1- Liaison parallèle :**

Dans ce type de liaison, tous les bits d'un mot sont transmis simultanément. Ce type de transmission permet des transferts rapides mais reste limitée à de faibles distances de transmission à cause du nombre important de lignes nécessaires (coût et encombrement) et des problèmes d'interférence électromagnétique entre chaque ligne (fiabilité). La transmission est cadencée par une horloge.

### **Exemple : Bus PCI**

### **2- Liaison série :**

Dans ce type de liaison, les bits constitutifs d'un mot sont transmis les uns après les autres sur un seul fil. Les distances de transmission peuvent donc être plus beaucoup plus importantes mais la vitesse de transmission est plus faible. Sur des distances supérieures à

Quelques dizaines de mètres, on utilisera des modems aux extrémités de la liaison.

La transmission de données en série peut se concevoir de deux façons différentes :

- **En mode synchrone**, l'émetteur et le récepteur possède une horloge synchronisée qui cadence la transmission. Le flot de données peut être ininterrompu.
- **En mode asynchrone**, la transmission s'effectue au rythme de la présence des données. Les caractères envoyés sont encadrés par un signal start et un signal stop.



## EXERCICE D'APPLICATION :

- 1- Quelle est la différence entre les échanges de données programmées et le DMA ?
- 2- Associez chaque mot de la liste suivante avec une des définitions proposées

a- Port série	b- Mémoire morte
c- Horloge	d- Transistor
e- Bit	f- CPU
g- Puce	h- Cache
i- Unité Centrale de traitement	j- Chiffre hexadécimal
k- Bus	l- Unité Arithmétique et Logique
m- Circuit intégré	n- Microprocesseur
o- RAM	p- Souris
q- Système d'exploitation	r- Unité de Contrôle
s- Clavier	t- Carte Mère
u- Imprimante	v- Pascal
w- Port parallèle	x- Mémoire vive
y- Registre	

1. Dispositifs d'entrée de données
2. Périphérique de sortie
3. Circuit spécialisé comportant une petite quantité de mémoire ultra rapide pour rendre les informations fréquemment demandées immédiatement accessibles au processeur.
4. Circuit intégré couramment utilisé pour construire la mémoire centrale.
5. Ensemble de circuits gravés sur une plaque de silicium.
6. Composante du microprocesseur qui indique quoi faire au bus, à la mémoire, aux dispositifs d'E/S en fonction des instructions du programme exécuté.
7. Chiffre en base 16
8. Unité d'information

9. Type de mémoire interne volatile, lisible et réinscriptible, dont chaque cellule est directement accessible
10. Langage de programmation
11. Cristal de quartz vibrant à une fréquence déterminée, produisant des signaux périodiques qui servent à synchroniser les tâches d'un microprocesseur.
12. Programme nécessaire à la gestion des ressources en matériel et logiciel d'un ordinateur.
13. Mémoire rapide à accès direct, Random Access Memory.
14. Dispositif d'E/S du micro-ordinateur accessible par un programme qui transporte simultanément les 8 bits d'un mot au travers d'un ensemble de conducteurs. Il peut connecter l'imprimante au micro-ordinateur.
15. Principale carte de circuits imprimés dans un ordinateur.
16. Dispositif d'E/S du micro-ordinateur accessible par un programme dans lequel les données sont transmises bit après bit de manière asynchrone. Il peut connecter le modem au micro-ordinateur.
17. Unité principale de traitement d'un ordinateur, généralement contenue dans un circuit intégré unique.
18. Ensemble des lignes transportant les signaux qui permettent au microprocesseur de communiquer avec ses mémoires internes et ses périphériques.
19. Petite mémoire contenue dans le microprocesseur destiné à stocker de manière très temporaire un certain nombre d'informations comme les résultats intermédiaires d'un calcul ou l'adresse de la prochaine instruction à exécuter.
20. Mémoire interne dont le contenu peut être lu mais non modifié.
21. Sorte de commutateur électronique servant à contrôler le passage du courant électrique. Sert à établir les niveaux de courant correspondant aux deux états numériques 0 et 1.
22. Partie du microprocesseur qui exécutent les calculs arithmétique (addition, soustraction, ...) et les opérations logiques (et, ou, ...).

## CORRECTION DES EXERCICES :

1.

Sans DMA	DMA
<p>1. Lecture bloc 2. Interruption du contrôleur 3. Ordre d'envoi du bloc en mémoire 4. <b>Transfert octet/octet en mémoire</b></p> <p> <b>4. Processeur occupé</b></p>	<p>1. Lecture bloc 2. Transfert bloc octet/octet mémoire 3. <b>Compteur=0 - Interruption</b></p> <p> <b>3. processeur libéré</b></p>
1) consommation temps CPU le temps de lire le bloc octet/octet en mémoire centrale.	1) pas d'occupation CPU le temps du transfert du bloc en mémoire.
2) CPU et contrôleur occupés le temps du transfert du bloc en mémoire.	2) Le transfert en mémoire ne se fait qu'après lecture du bloc complet ceci évite la perte de données arrivant à un rythme constant du disque.
3) La lecture du bloc suivant n'est lancée qu'après ce transfert.	

2.

Définitions	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Mots	s	u	h	m	g	r	j	e	x	v	c	q	o	w	t	a	i, f, k, y	b	d	l		
	p															n						